Securitatea în rețele de calculatoare

Implementarea securității într-o rețea de calculatoare cuprinde trei aspecte importante: confidențialitatea, integritatea şi disponibilitatea.

Confidențialitatea reprezintă calitatea unei rețele de a asigura accesul la informație doar persoanelor autorizate.

Integritatea garantează faptul că informația nu a fost modificată de persoane neautorizate.

Disponibilitatea poate fi definită ca timpul în care rețeaua de calculatoare şi resursele din cadrul ei sunt operaționale.

Pentru fiecare din aceste aspecte ale securității rețelelor de calculatoare există atacuri, astfel încât securizarea unei rețele de calculatoare trebuie să implementeze fiecare din aceste aspecte.

Probleme:

- identificarea, autorizarea şi monitorizarea activității utilizatorilor
- securizarea perimetrului rețelei
- asigurarea confidențialității şi integrității datelor
- monitorizarea rețelei
- managementul echipamentelor şi infrastructurii de securitate.

- soluții : firewall-uri, VPN-uri, sisteme de detecție a intruziunilor, etc.
- definirea unei politici de securitate

- O politică de securitate este ``o definire formală a regulilor după care persoanele care au acces la bunurile tehnologice şi informatice ale organizaţiei trebuie să le respecte".
    - o politică de securitate specifică ce, cine şi în ce condiţii se pot accesa anumite resurse ale organizaţiei
    - politica de securitate trebuie implementată pentru a ne asigura că este respectată.

Tipuri de atacuri şi vulnerabilităţi

Există două cauze majore ce pot constitui ameninţări pentru o reţea de calculatoare, chiar după ce a fost implementată o politică de securitate corectă:

1. vulnerabilităţi (probleme cauzate de tehnologie) şi
2. configurare necorespunzătoare.

Vulnerabilităţile sunt probleme ale sistemelor de operare, protocoalelor TCP/IP, dispozitivelor de reţea prin care un atacator poate accesa reţeaua fără a respecta politica de securitate implementată.

Chiar dacă vulnerabilităţile sunt problemele cele mai grave şi mai greu de controlat, trebuie însă notat că cele mai multe probleme apar datorită *configurării incorecte* sau definirii unei politici de securitate necorespunzătoare.

Atacurile asupra unei rețele de calculatoare pot fi clasificate în :

1. *atacuri interne* sau *externe* şi
2. *atacuri structurate* sau *nestructurate*.

Atacurile externe sunt efectuate din afara organizației (din punctul de vedere al rețelei). Atacurile interne sunt efectuate din rețeaua organizației.

Atacurile nestructurate sunt atacurile care sunt inițiate de indivizi neexperimentați ce utilizează exploit-uri disponibile pe Internet. *Exploit-urile* sunt programe ce exploatează vulnerabilitățile pentru a ocoli politica de securitate implementată într-o rețea.

Atacurile structurate sunt inițiate de indivizi mult mai bine motivați şi cu cunoştințe tehnice competente. Aceşti indivizi cunosc vulnerabilități de sistem şi le pot folosi pentru a căpăta acces în rețea, pot detecta noi vulnerabilități de sistem şi pot dezvolta cod şi scripturi pentru a le exploata.

Un atac trece în general prin trei faze:

1. faza de recunoaştere,
2. faza de obținere a accesului - formată din două etape: una în care atacatorul obține acces în cadrul rețelei pe una din maşinile din rețea prin *exploit-uri de la distanță* şi faza în care, dacă este cazul, obține acces privilegiat pe maşina respectivă cu ajutorul unor *exploit-uri locale.*
3. faza în care sistemele compromise sunt folosite pentru a ataca alte rețele. (eventual)

1. Recunoaşterea

- defineşte procesul prin care un atacator descoperă maparea sistemelor, a serviciilor şi vulnerabilităţilor în reţea. În această fază atacatorul strânge informaţii şi, de cele mai multe ori, această fază precede un atac efectiv.

Într-o primă fază atacatorul foloseşte utilitare precum *nslookup* sau *whois* pentru a descoperi spaţiul de adrese alocate organizaţiei ţintă. Apoi face un *ping sweep* încercând să determine care din adresele de IP sunt alocate şi care din sisteme sunt pornite. Se foloseşte apoi un *port scanner* pentru a determina ce servicii sunt active. Acest utilitar funcţionează pe principiul că fiecare serviciu (web, ftp, etc) are alocat un port. Utilitarul trimite pachete SYN către maşina atacată pe portul corespunzător serviciului care se încearcă a fi detectat. Dacă maşina atacată rulează serviciul, sistemul de operare va trimite un pachet de tipul SYN, ACK pentru a începe negocierea unui canal de comunicaţie. Acest utilitar poate de asemenea să detecteze şi tipul sistemului de operare, cel mai adesea datorită modului în care unele din sistemele de operare generează numere de secvenţă pentru pachetele TCP. După determinarea serviciilor şi sistemului de operare, atacatorul încearcă să obţină versiunea sistemului de operare şi versiunile serviciilor rulate. Acest lucru se poate face prin conectarea cu utilitare de gen *telnet* pe portul serviciului respectiv şi examinarea mesajele afişate. Pe baza acestor informaţii atacatorul poate determina ce vulnerabilităţi există şi ce sisteme poate ataca.

O altă modalitatea de recunoaştere a resurselor o reprezintă aşa numitul proces de *packet sniffing*. El este folosit mai ales în reţelele în care mesajele ajung la toată lumea conectată la mediu, ca în cazul Ethernet atunci când se foloseşte un hub. Datorită acestui lucru, tot traficul dintr-o astfel de reţea poate fi analizat. În mod normal, placa de reţea nu va prelua din mediu decât pachetele destinate staţiei respective sau pachetele de broadcast (la nivel 2). Dacă există privilegii suficiente, se poate configura placa de reţea astfel încât să preia toate pachetele ce circulă pe mediu, prin setarea acesteia în *promiscuous mode*. Pachetele astfel captate pot fi procesate cu diverse utilitare şi pe baza lor se pot mapa adrese, versiuni de sisteme de operare sau servicii. Mai mult, folosind utilitare de packet sniffing se pot recepţiona chiar şi date importante care ar trebui sa aibă un caracter privat, cum ar fi parole, numere de cărţi de credit, informaţii confidenţiale, etc. O falsă soluţie pentru a preveni problemele ce apar atunci când se folosesc utilitare de packet sniffing este să se folosească un switch ca metodă de interconectare în loc de un hub. Deşi reduce probabilitatea ca un atacator să poată intercepta pachetele, nu este o metodă sigură. Sunt cunoscute metode prin care un switch poate fi păcălit să trimită pachete şi către alte porturi (şi implicit calculatoare), nu doar către portul destinaţie. Aceste metode poartă numele de *ARP poisoning*.

Metode eficiente de protecţie împotriva unui atac de recunoaştere sunt:

- folosirea unui firewall, pentru a bloca încercările de recunoaştere,
- folosirea doar a unor protocoale sigure care nu trimit datele în clar, ci criptate, sau
- folosirea criptării pentru protocoalele nesigure prin tunelare.

2. Obținerea accesului

Una dintre cele mai sigure metode de obținere a accesului privilegiat este a *sparge parola*. Acest lucru presupune că atacatorul are deja acces pe o maşină din rețea şi doreşte acces privilegiat (root, Administrator). Deşi un atac ``brute force'' nu are cum să dea rezultate decât pe sistemele la care parolele sunt limitate la 6-7 caractere, există atacuri care se folosesc de unele particularități ale parolelor. S-a observat că majoritatea parolelor folosite se încadrează în anumite categorii, pentru a putea fi uşor ținute minte. Din această cauză există utilitare de spart parole bazate pe dicționare (*Jack The Ripper*).

Altă metodă de exploit-uri de la distanță este *deturnarea unei conexiuni TCP* (*TCP session hijack*). Ea constă în aşteptarea ca un utilizator să se logheze pe sistem ce doreşte să fie atacat, şi apoi în trimiterea de pachete către portul pe care rulează serviciul, luând locul utilizatorului care s-a autentificat. Această metodă se foloseşte dacă se pot prezice numerele de secvență dintr-un pachet TCP. O variantă de deturnare de conexiuni TCP este *man in the middle attack*. În acest caz, atacatorul trebuie să aibă acces la o maşină pe care trece traficul dintre două entități A şi B. În acest caz, atacatorul interceptează cererea de conexiune de la A la B şi răspunde lui A, stabilind cu A o conexiune. Apoi stabileşte şi cu B o conexiune. Toate datele trimise de A vor fi apoi trimise lui B şi

invers. Astfel atacatorul are acces la convorbirea dintre A şi B, chiar dacă traficul este criptat din punctul de vedere al lui A şi B.

IP spoofing este o metodă de atac, dar poate fi folosită şi pentru a ascunde identitatea atacatorului sau pentru a lansa atacuri. Prin acest atac, pachetele TCP/IP sunt manipulate, falsificând adresa sursă. În acest mod atacatorul poate căpăta acces atribuindu-şi o identitate (adresa de IP) care are autorizare să acceseze resursa atacată. Datorită falsificării adresei sursă a pachetului IP, atacatorul nu poate stabili decât o comunicație unidirecțională (presupunând că nu este prezent în rețeaua locală a maşinii atacate). Acest lucru face protocolul TCP nesusceptibil pentru asemenea atacuri. Există însă numeroase servicii UDP care pot fi exploatate cu acest tip de atac.

Viruşii pot constitui metode de atac, atunci când poartă cu ei troieni. *Troienii* sunt programe simple, care deschid uşi de acces pe sistemele infectate de virus.

O metodă diferită față de cele discutate până acum o reprezintă *ingineria socială*. Ea constă în aflarea de informații esențiale direct de la utilizatori.

3. *Denial of Service (DoS)*

faza în care sistemele compromise sunt folosite pentru a ataca alte rețele

Atacurile de tipul *denial of service (DoS)* opresc sau încetinesc foarte mult funcționarea unor rețele, sisteme sau servicii. Ele sunt cauzate de un atacator care doreşte să împiedice accesul utilizatorilor la resursele atacate. Atacatorul nu are nevoie să fi căpătat înainte acces pe calculatorul pe care doreşte să efectueze atacul. Există multe posibilități prin care un atac DoS se poate manifesta. Efectul însă este acelaşi: se împiedică accesul persoanelor autorizate de a folosi serviciile de pe sistem prin utilizarea la maxim a resurselor sistemului de către atacator.

- Exemplu de atac DoS local este un program care creează procese la infinit. Acest lucru va duce în cele din urmă la încetinirea sistemului, pentru că existând un număr foarte mare de procese create de atacator, probabilitatea ca acestea să se execute va fi foarte mare, iar procesele celorlalți utilizatori nu mai apucă să se execute.

- Un alt tip de atac DoS local posibil este crearea unui număr limitat de procese (pentru că majoritatea sistemelor de operare moderne limitează numărul maxim de procese pe care un utilizator le poate crea), care alocă zone de memorie de dimensiuni mari şi care accesează aceste zone aleator. Ideea acestui atac este de a forța sistemul de operare să lucreze cu swap-ul, încetinindu-l.

- Există şi atacuri DoS de la distanță - se bazează pe vulnerabilități ale SO sau aplicațiilor. Un exemplu este atacul cu date *out of band*, conceput pentru sistemele de operare Windows 95 şi NT. Acest atac va determina sistemul de operare să se blocheze sau să se reseteze. Datele *out of band* sunt date care nu fac parte din fluxul normal de date schimbat între doi socketi. Acest tip de date sunt trimise doar în cazuri speciale şi au prioritate față de datele normale. Un exemplu de situație în care datele *out of band* sunt folositoare poate fi următorul: presupunem că avem o aplicație client - server ce implementează un protocol similar cu telnet. Astfel, între client şi server se trimit comenzile, respectiv outputul acestora. Pentru a suporta dimensiuni variabile ale ecranului la client, în momentul în care acesta redimensionează fereastra se trimite serverului un mesaj *out of band* în care se specifică noile dimensiuni ale ecranului.

- Atacul *Ping of Death* foloseşte pachete IP modificate care indică faptul că pachetul are mai multe date decât are de fapt. Acest atac determină blocarea sau resetarea maşinii pe sistemele care nu verifică acest lucru.

Atacuri DoS distribuite

Atacurile DoS distribuite sunt astfel concepute încât să satureze lărgimea de bandă pe legătura ce conectează rețeaua la Internet cu pachete de date trimise de atacator, astfel încât pachetele legitime nu mai pot fi trimise. Pentru a realiza acest lucru un atacator se foloseşte, direct sau indirect, de mai multe sisteme .

- Un exemplu de astfel de atac este *Smurf*. Acesta începe prin a trimite un număr mare de mesaje ICMP de tip echo-request (sau ping) către adrese de broadcast, sperând că aceste pachete vor fi trimise unui întreg segment de rețea. De asemenea aceste pachete sunt falsificate pentru a avea ca adresă sursă adresa sistemului țintă. Dacă pachetul trece de dispozitivul de rutare, el va fi recepționat de către toate stațiile de pe un segment de rețea. Acestea vor răspunde cu un pachet de tip echo-reply către adresa falsă din pachet. Astfel, stațiile vor genera trafic către adresa specificată de atacator. Acest tip de atac poate fi uşor contracarat dacă pe ruter se dezactivează rutarea pachetelor de broadcast direcționat.

- Un alt tip de atac distribuit se poate realiza cu pachetul de utilitare *TFN2K* (*Tribe Flood Network 2000*). Atacul TFN are capacitatea de a genera pachete de IP cu adresa sursă falsificată. În prealabil însă, sistemele de pe care se face atacul trebuie să fi fost instalate cu aceste utilitare. Acest lucru se face în primul pas, când se atacă stațiile (denumite drone-uri). Un TFN master poate apoi comanda drone-urile cauzând atacuri DoS distribuite.

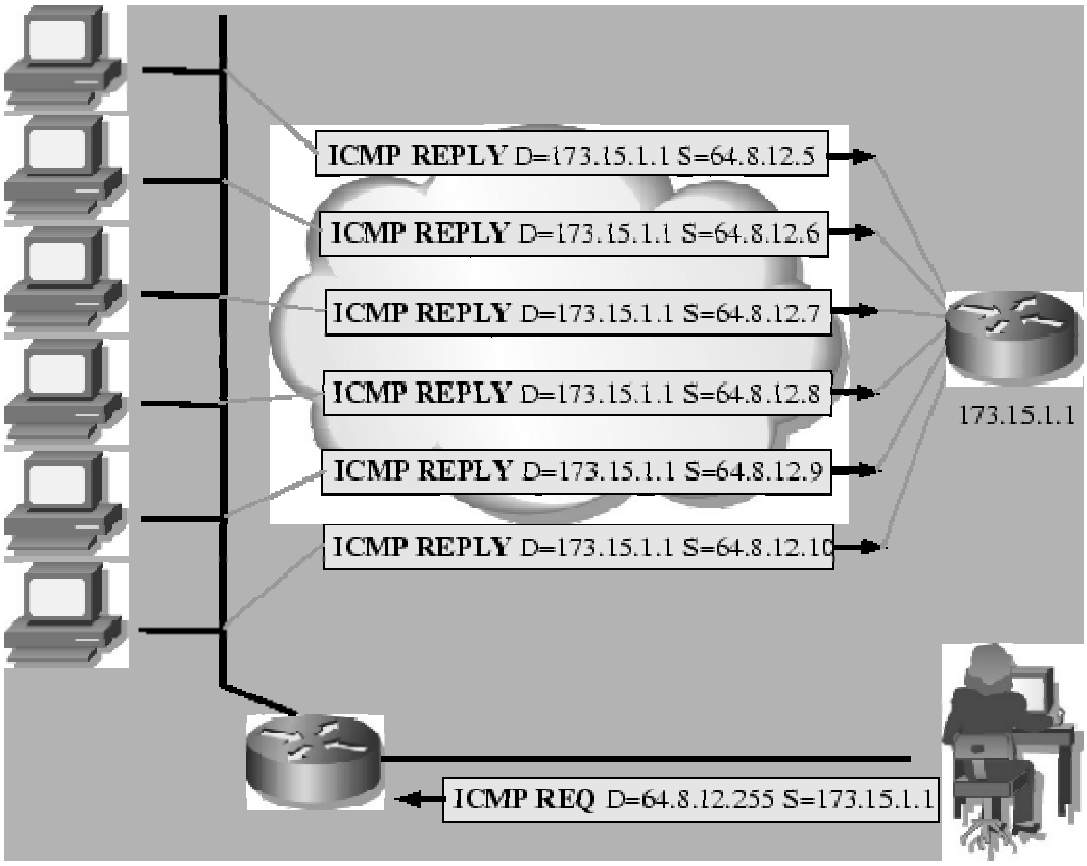

Figura : Exemplu de atac Smurf

Soluții pentru implementarea securității

- asigurarea securității perimetrului rețelei se face cu un *firewall,* dispozitiv special de rețea.
- Monitorizarea rețelei se face cu un *IDS (Intrusion Detection System),* alt dispozitiv special de rețea.
- Păstrarea confidențialității şi integrității datelor într-un mediu ostil se face cu tehnologii *VPN (Virtual Private Network)*.
- Pentru identificarea, autorizarea şi monitorizarea activității *(accounting)* utilizatorilor într-un mod centralizat se folosesc *protocoale* precum *RADIUS (Remote Authentication Dial-In User Service)* sau *TACACS (Terminal Access Controller Access Control System)*.

1. Firewall-ul

- firewall-ul este un sistem sau un grup de sisteme care implementează politica de acces între două sau mai multe rețele.

- Firewall-urile pot fi clasificate în patru mari clase:

  1. *firewall-uri dedicate*,
  2. *firewall-uri de rutere*,
  3. *firewall-uri de server* şi
  4. *firewall-uri personale.*

Firewall-urile dedicate sunt maşini ce rulează un sistem de operare special conceput pentru filtrarea de pachete şi translatarea de adrese.

Firewall-urile de rutere reprezintă de fapt software special care rulează pe rutere. Ruterul este astfel integrat cu facilități de firewall.

Firewall-urile de server sunt implementate, în general, ca un software adițional peste un sistem de operare de rețea (Linux, NT, Win2K, Novell, UNIX). Exemple de astfel de pachete software sunt: *Netfilter*, *Microsoft ISA Server*, *Novell Border Manager*. Din cauză că rulează software peste un sistem de operare de uz general, aceste tipuri de firewall-uri nu se descurcă la fel de bine în situații de încărcare mare ca un firewall dedicat.

Firewall-urile personale sunt instalate pe calculatoarele personale. Ele sunt concepute pentru a preveni atacuri doar asupra calculatorului pe care rulează. Este important de reținut că aceste tipuri de firewall-uri nu sunt optimizate pentru rețele întregi de calculatoare.

Principalele mecanisme prin care un firewall asigură protecția rețelei sunt : filtrarea de pachete şi translatarea de adrese.

Filtrarea de pachete

Filtrarea de pachete este procesul prin care firewall-ul lasă să treacă în rețeaua locală doar anumite pachete, pe baza unor reguli. Filtrarea de pachete este folosită pentru a proteja o rețea de atacuri din exterior (Internet) şi se realizează la nivelurile OSI 3 şi 4.

Regulile de filtrare sunt formate dintr-o parte care identifică pachetul şi o parte care specifică cum să se trateze pachetul.

În partea de identificare se poate specifica *adresa sursă, adresa destinație, adresa de rețea sursă, adresa de rețea destinație, protocolul (TCP, UDP, ICMP), portul sursă sau destinație (doar pentru TCP sau UDP), tipul mesajului (pentru ICMP), interfața de intrare sau ieşire, adresele de nivel doi*, etc.

Partea de tratare a pachetului specifică ce anume trebuie făcut cu pachetele identificate de regulă. Pentru filtrare există în general 3 posibilități de tratare: *acceptare, ignorare sau respingere*. În cazul acceptării pachetul este lăsat să treacă. În cazul ignorării pachetului nu este lăsat să treacă şi nu se trimite notificare către sursă. În cazul respingerii pachetul nu este lăsat să treacă, dar se trimite notificare către sursă (un mesaj ICMP al cărui tip poate fi, în unele implementări, ales de cel care construieşte regula; de cele mai multe ori se foloseşte un mesaj ICMP de tip *port-unreachable*).

Translatarea de adrese

Translatarea de adrese sau NAT este procesul prin care un ruter modifică adresele sursă (SNAT) sau destinație (DNAT) din anumite pachete care trec prin ruter pe baza unor reguli.

Putem considera că translatarea adreselor este o funcție definită pe o mulțime de adrese (A) cu rezultate într-o altă mulțime de adrese (B). Astfel, fiecare pachet cu o adresă sursă sau destinație (după cum este specificat în regulă) din mulțimea A va fi înlocuită cu o adresă din mulțimea B.

Se spune că avem o translatare de adresă statică dacă mulțimile A şi B sunt fiecare formate dintr-un singur element. În caz contrar avem o translatare de adrese dinamică.

Avantajul folosirii translatării de adrese dinamice constă în faptul că se poate folosi o partajare a adreselor rutabile disponibile organizației. Astfel, calculatoarelor din rețeaua locală li se alocă adrese private, iar ruterul va face o translatare de adrese dinamice din mulțimea de adrese private în mulțimea de adrese publice alocate organizației. Se observă însă că această abordare permite ca doar Card(B) calculatoare din rețeaua locală să aibă conversații TCP sau UDP cu Internetul. Alt avantaj al folosirii translatării de adrese este acela că se ascunde astfel exteriorului maparea reală de adrese.

Translatarea de adrese statică se foloseşte atunci când în rețeaua locală avem un server pe care dorim să îl accesăm din exterior. În acest caz se face o mapare unu la unu între adresa din interior şi cea din exterior.

O metodă mai avansată de translatare de adrese o reprezintă *PAT (Port Address Translation),* uneori denumită şi *NAT overloaded.* Această metodă permite un număr de aproximativ 64000 de conversaţii simultane de la orice host intern către exterior cu o singură adresă externă. Implementarea înlocuieşte pachetul din reţeaua locală cu adresa sursă S, adresa destinaţie D, portul sursă P, portul destinaţie Q, cu altul nou ce va avea adresa sursă M (adresa ruterului), adresa destinaţie D, portul sursa K. Portul destinaţie nu se schimbă. De asemenea se memorează asocierea (S,P) - K. Dacă un pachet ajunge pe ruter din exterior, având adresa destinaţie M, adresa sursă Q şi portul destinaţie K, atunci acest pachet va fi înlocuit cu un altul cu adresa destinaţie S, adresa sursă Q, portul destinaţie P şi va fi trimis în reţeaua locală. Portul sursă nu se schimbă.

Un caz special al PAT îl reprezintă redirectarea. În acest caz se va înlocui pachetul primit din reţeaua locală având adresa sursă S, adresa destinaţie D, portul P cu un altul având adresa sursă S, adresa destinaţie M (adresa ruterului), portul R (portul în care se face redirectarea, specificat de utilizator). Redirectarea este în general folosită pentru a implementa un proxy transparent, caz în care pe ruterul M portul R ascultă un proxy configurat pentru proxy transparent.

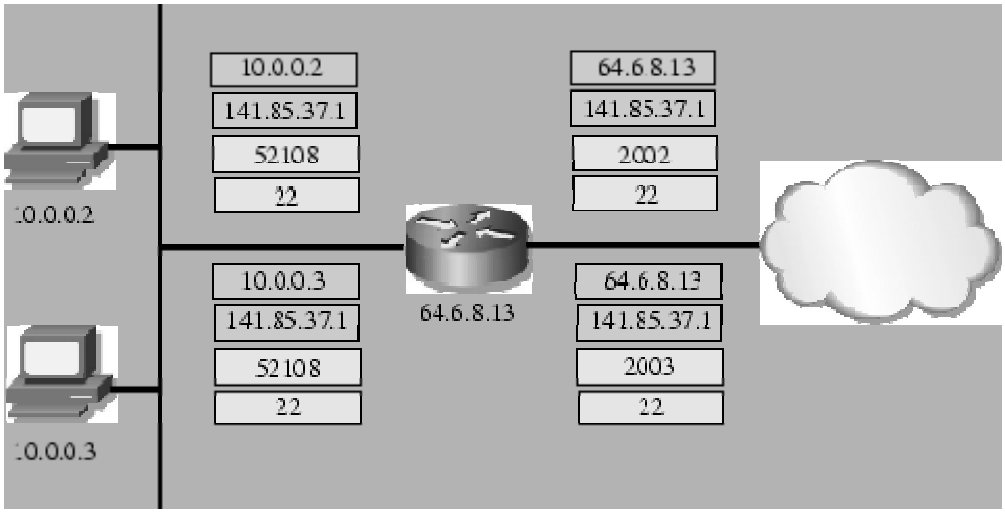

Figura : NAT overloaded

Filtrare de pachete şi translatare de adrese avansată

Anumite protocoale, datorită felului în care sunt concepute, pot să nu funcţioneze corect atunci când clientul şi serverul sunt separate de un firewall care filtrează pachete sau implementează PAT. În general, pentru astfel de protocoale clientul şi serverul negociază un port pentru client şi apoi serverul iniţiază o conexiune către client pe portul negociat. Din această cauză, implementarea unui mecanism de filtrare care să permită funcţionarea acestui protocol, dar să protejeze staţia de atacuri din exterior, se complică extrem de mult. La fel stau lucrurile şi pentru PAT.

Exemplu de astfel de protocoale : FTP.

Protocolul FTP foloseşte două porturi:

- portul de date (*ftp-date*) şi
- portul de comenzi (*ftp-comenzi*).

La conectarea la server, clientul iniţiază o conexiune către portul ftp-comenzi. În momentul în care clientul doreşte transferul unui fişier el va pregăti un port pe care va asculta cereri de conexiuni de la server, după care va trimite pe canalul de comenzi un mesaj în care i se cere serverului fişierul de transferat şi în care îi trimite serverului portul pe care clientul ascultă. Serverul va iniţia apoi o conexiune de pe portul ftp-date către portul specificat de client. Aceste este modul de funcţionare normal pentru protocolul FTP. Clientul poate fi însă configurat să ceară de la server un mod de lucru pasiv, în care doar clientul iniţiază conexiuni.

Fie o situație în care dorim să utilizăm filtrarea de pachete pentru a proteja rețeua locală de atacuri din exterior. Astfel, pe firewall nu vom permite ca stațiile din exterior să inițieze conexiuni către stațiile din interior. Din acest motiv, în momentul în care o stație locală va încerca să transfere un fişier în mod normal de pe un server de FTP, firewall-ul va bloca încercarea de deschidere a unei conexiuni a serverului către client. Cum portul este negociat de client, problema apărută nu se poate rezolva foarte simplu.

Singura soluție posibilă este ca firewall-ul să analizeze toate pachetele schimbate de client şi server şi să identifice portul negociat. Cu această informație va putea apoi permite stabilirea conexiunii inițiate de server cu clientul. Folosirea unei astfel de abordări este denumită *stateful inspection* sau *connection tracking*.

2 Sisteme de detecție a intruziunilor

Sistemele de detecție a intruziunilor - *Intrusion Detection Systems (IDS)* au abilitatea de a detecta atacurile împotriva unei rețele. Aceste sisteme identifică, opresc şi semnalează atacurile asupra resurselor rețelei.

Există două tipuri de sisteme de detecție a intruziunilor:

- pentru stații şi
- pentru rețele.

Un *HIDS* (*Host based IDS*) sau un sistem de detecție a intruziunilor pentru stații înregistrează atât operațiile efectuate, cât şi utilizarea resurselor sistemului. Un avantaj al HIDS este faptul că el poate preveni atacuri necunoscute. De exemplu un HIDS poate monitoriza accesul la fişiere şi reacționa când un atacator încearcă să şteargă fişiere critice. Chiar dacă tipul atacului este nou şi nu poate fi recunoscut de un *NIDS* (*Network based IDS*) un HIDS poate sesiza atacul.

Cea mai simplă formă de HIDS este pornirea proceselor de logare pe sistem. O astfel de metodă se spune că este pasivă. Dezavantajul acestei metode este faptul că necesită multe ore de muncă din partea administratorului pentru a analiza logurile. Sistemele HIDS curente folosesc agenți software care sunt instalați pe fiecare maşină şi care monitorizează activ sistemul (pot reacționa dacă detectează atacuri). Atunci când HIDS-ul este configurat să răspundă activ, el va opri serviciile de rețea pentru a preveni eventuale pagube şi a putea analiza exact atacul. Un exemplu de sistem de detecție a intruziunilor este *Linux Intrusion Detection System (LIDS)*.

NIDS-urile sunt dispozitive de inspectare a traficului şi acționează prin colectarea de date de la senzori amplasați în rețea. NIDS-urile captează si analizează traficul ce traversează rețeaua. Avantajul folosirii unui NIDS este faptul că nu trebuie aduse modificări pe stații. În schimb, datorită faptului că la baza NIDS-urilor stă detecția bazată pe semnături ale atacurilor, el nu poate opri sau detecta atacuri noi.